

IN THE SPECIFICATION:

Please replace paragraph [0010] with the following:

[0010] In another form, the invention includes a method of blurring a digital video image having a plurality of pixels, each having a value specifying color data. The method comprises identifying a particular pixel of the plurality of pixels; selecting  $2^N$  pixels from the plurality of pixels, where N is a positive integer; ~~right shifting by N bits the bits~~ masking the least significant N bits of the value of each selected pixel to get a modified value of each selected pixel; dividing the modified value by  $2^N$  to get a divided value for each selected pixel; summing the divided values to get a total value; and replacing the value of the particular pixel with the total value whereby the particular pixel has a blurred value.

Please replace paragraph [0030] with the following:

[0030] As another example, the particular pixel value may be used in the process of generating the blurred pixel value. For example, as shown in FIG. 1 the value of pixels 108, 110 and 112 could each be divided by one-third and added to determine the blurred pixel value which would be substituted for the center pixel 110. However, this ~~would~~ could not be viewed as a fast implementation ~~as the shift operation (for dividing)~~ as you would not be ~~in a convenient form for further processing~~ able to accomplish the divide in a single operation.

Please replace paragraph [0035] with the following:

[0035] As noted above with regard to one embodiment, the particular pixel would be identified and  $2^N$  pixels from the pixels surrounding the particular pixel would be selected, where N is a positive integer. In one embodiment N equals 1, 2 or 3. One advantage of this embodiment is that the entire operation may be accomplished by a small number of fast machine operations (sometimes referred to as "a single line of computer code machine code instruction"). The pixels are averaged without having to separate each channel component. This is because, as noted above, in binary the values of the pixels can easily be divided by two, four or eight simply by ignoring (masking) one or more of the least significant bits and by right shifting the binary value. For example, an eight bit binary value for one of the components of the color data of a particular pixel can be divided by ignoring the three least significant bits and by right shifting to the right three places.

Please replace paragraph [0037] with the following:

[0037] Finally, the same method may applied on all 8 surrounding pixels (FIG. 6), by ignoring their last 3 LSBs, and averaging them (right-shifting them by 3 is equivalent to dividing them by 8) to obtain a blur approximation (23 operations per pixel):

```

NewValueOfCurrentPixel = (PixelOnTheLeft&0xF8F8F8F8)>>3 +
(PixelOnTheRight&0xF8F8F8F8)>>3 +
(PixelOnTheTopLeft&0xF8F8F8F8)>>3 +
(PixelOnTheTop&0xF8F8F8F8)>>3 +
(PixelOnTheTopRight&0xF8F8F8F8)>>3 +
(PixelOnTheBottomLeft&0xF8F8F8F8)>>3 +
(PixelOnTheBottomLeft&0xF8F8F8F8)>>3 +
(PixelOnTheBottom&0xF8F8F8F8)>>3 +
(PixelOnTheBottomRight&0xF8F8F8F8)>>3

```